2015 IEEE 14th International Conference on Machine Learning and Applications

Deep Neural Networks: A Case Study for Music Genre Classification

Arjun Raj Rajanna*, Kamelia Aryafar†, Ali Shokoufandeh† and Raymond Ptucha‡
*Electrical Engineering Department
Rochester Institute of Technology, Rochester, NY 14623
† Computer Science Department
Drexel University, Philadelphia, PA 19104
‡Computer Engineering Department
Rochester Institute of Technology, Rochester, NY 14623

***Abstract*—Music classification is a challenging problem with many applications in today's large-scale datasets with Gigabytes of music files and associated metadata and online streaming services. Recent success with deep neural network architectures on large-scale datasets has inspired numerous studies in the machine learning community for various pattern recognition and classification tasks such as automatic speech recognition, natural language processing, audio classification and computer vision. In this paper, we explore a two-layer neural network with manifold learning techniques for music genre classification. We compare the classification accuracy rate of deep neural networks with a set of well-known learning models including support vector machines (SVM and ℓ_1-SVM), logistic regression and ℓ_1-regression in combination with hand-crafted audio features for a genre classification task on a public dataset. Our experimental results show that neural networks are comparable with classic learning models when the data is represented in a rich feature space.**

I. Introduction

Automatic music classification is a fundamental problem for music indexing, content-based music retrieval, music recommendation and online music distribution. Various large-scale datasets of Gigabytes of music information along with metadata and online music streaming services such as Spotify [1] and Pandora [2] are available. Due to enormity of these datasets, scalable machine learning models that can categorize music information by different criteria such as artist, genre and music similarity are required. Numerous methods have been developed over the years to efficiently classify music information, but the hurdles remain [16]. In this paper we explore scalable approaches to music classification by genre on a public dataset and examine the challenging aspects of this problem.

Music classification presents two main challenges: 1) preprocessing and extraction of meaningful audio features; and 2) the choice of a learning model. The music information retrieval community has developed numerous hand-crafted feature vectors to represent input music data to boost the classification and retrieval accuracy. Audio signals are often characterized by signal-based features [40], [8], [15] including timbre, harmony, and rhythm. These features can be described as short-time and long-time audio features. The short-time audio features are mainly derived from short segments of the signal spectrum and include spectral centroids, Mel-frequency cepstral coefficients (MFCC) [44], chroma features [13] and octave based spectral contrast (OSC) [23]. These short-time feature vectors are often extended to include the whole signal length by a sliding window such as Hann function or Hamming window [14]. The long-time audio features are mainly based on variation of spectral or beat information over a long segment of the audio signal. Typical examples of long-time audio features include Daubechies wavelet coefficients histogram (DWCH) [28], octave-based modulation spectral contrast (OMSC), low-energy and beat histogram [44].

Signal-based audio features often lack high-level abstractions that are critical for characterizing music genre [33]. High-level features including distance-based [11] features and statistical [30] features have been proposed to address this problem. There have been several attempts in combining signal-based and high-level audio features for improving the classification accuracy [30], [1], [35] on public datasets. Textual features extracted from lyrics, web and social tags are also utilized as another source of higher-level information in semantic space for music classification and retrieval [38], [45], [34]. There have also been several efforts in combining these features with a learning model such as support vector machines (SVM) [16], logistic regression, ℓ_1-SVM [7], [3], [2], naive Bayes classifiers and k-nearest neighbors (k-NN) to boost the classification accuracy [16]. While the feature detection and preprocessing algorithms for music datasets appear to be maturing, the automatic music classification remains an active area of research. Scalable classification algorithms are often required to validate the results on larger, well-defined, but more diverse datasets.

Classification of music information can be done with and without input labels. If no labels are provided the classification problem is redocued to a clustering. Spectral clustering [39], k-means clustering and other models can be used to clusters input signals by similarity without labels. Given a set of labeled training examples a classifier can provide a class label for unseen test examples (a predictive perspective) or describe existing patterns in a dataset (descriptive prespective). The classification performance, convergence time, robustness, simplicity, interpretability, scalability and domain-dependent quality indicators are often the most important evaluation metrics for a classifier. Hand-crafted audio features in combination with SVMs [16], logistic regression, ℓ_1-SVM [7], [3],

[1] www.spotify.com

[2] http://www.pandora.com/

978-1-5090-0287-0/15 $31.00 © 2015 IEEE
DOI 10.1109/ICMLA.2015.160

naive Bayes classifiers and k-nearest neighbors (k-NN) have been widely explored for music classification by artist and genre [16], [4], [5], [6].

In recent years, deep learning architectures [12] such as deep neural networks [9], convolutional deep neural networks [25], deep belief networks [21] and recurrent neural networks [18] have been applied to fields like automatic speech recognition, natural language processing, audio recognition and bioinformatics where they have been shown to produce state-of-the-art results on various tasks and have won numerous contests in pattern recognition and machine learning. The success with deep network architectures on large-scale datasets has inspired music classification efforts using deep networks [19], [26]. Deep networks have recently been used to improve feature extraction on audio signals and boost the music classification accuracy [42]. These efforts pose an interesting question: Can neural networks outperform and-crafted features in combination with well-known classifiers such as SVMs? In this paper, we explore the answers to this question in an experimental settings on a public dataset.

This paper is organized as follows: In section II we present the outline of our proposed model. Preprocessing is presented in Section II-A. Section II-B explores various manifold learning and dimensionality reduction techniques that are used in our model. Section II-C explores the deep network architectures used in our experiments and the Rectified Linear Units (ReLUs) as activation functions. We present the comparative results of music genre classification on a sample dataset in Section III. Finally we conclude this paper in Section IV and propose future directions to enhance the empirical results.

II. Method

The input data to our music genre classification is a set of unprocessed (raw) audio signals in wav format. Figure 2 illustrates examples of the input signals from the *alternative, pop and rock* genre classes in frequency domain and the power spectrogram from the sample dataset. Our proposed method involves five successive sequences: preprocessing of audio signals to normalize and augment input data [3], feature extraction or spectrogram representation, manifold learning and dimensionality reduction techniques, classification model such as SVMs or DNNs and genre label prediction of unseen (test) examples. Figure 1 shows the stages of the proposed method.

A. Preprocessing

The preprocessing and representation of raw input audio signals vary among different experiments which will be described in Section III. The preprocessing steps on raw wav files can be divided into three main steps: data manipulation (augmentation, normalization, subsampling and etc), power sperctrogram representation and feature extraction. The first step in preprocessing includes subsampling all input signals by resizing to the minimum sample size across all dataset. In this paper, we have avoided zero padding for noise reduction and better accuracy rate for neural networks as described in Section III. The DNNs first utilize the raw audio features with resizing and the power spectrogram as a set of input. We have also experimented with wavelet filtering of resized audio signals along with DNNs to boost the classification accuracy. The SVMs and regression based classifiers use hand-crafted features (Mel-frequency cepstral coefficients (MFCCs) [44]) to enhance the classification accuracy rate.

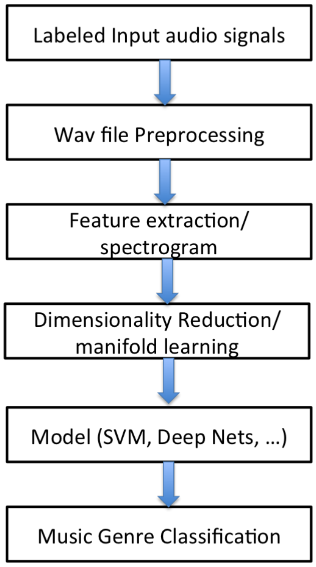

Labeled Input audio signals
Wav file Preprocessing
Feature extraction/ spectrogram
Dimensionality Reduction/ manifold learning
Model (SVM, Deep Nets, …)
Music Genre Classification

Fig. 1. The music classification method is illustrated.

1) Spectrogram: A spectrogram is a visual representation of the spectrum of the input signal frequencies as they vary with time and have been used to identify spoken words phonetically and to analyze the various calls of animals. Spectrograms are often created in one of two ways: approximated as a filterbank resulting from a series of bandpass filters (this was the only way before the advent of modern digital signal processing), or calculated from the time signal using the fast Fourier transform (FFT [4]). In this paper we use the FFT to generate the power spectrogram of input audio signal $x(t)$. The spectrogram of $x(t)$ can be estimated by computing the squared magnitude of the STFT of the signal $x(t)$, as follows:

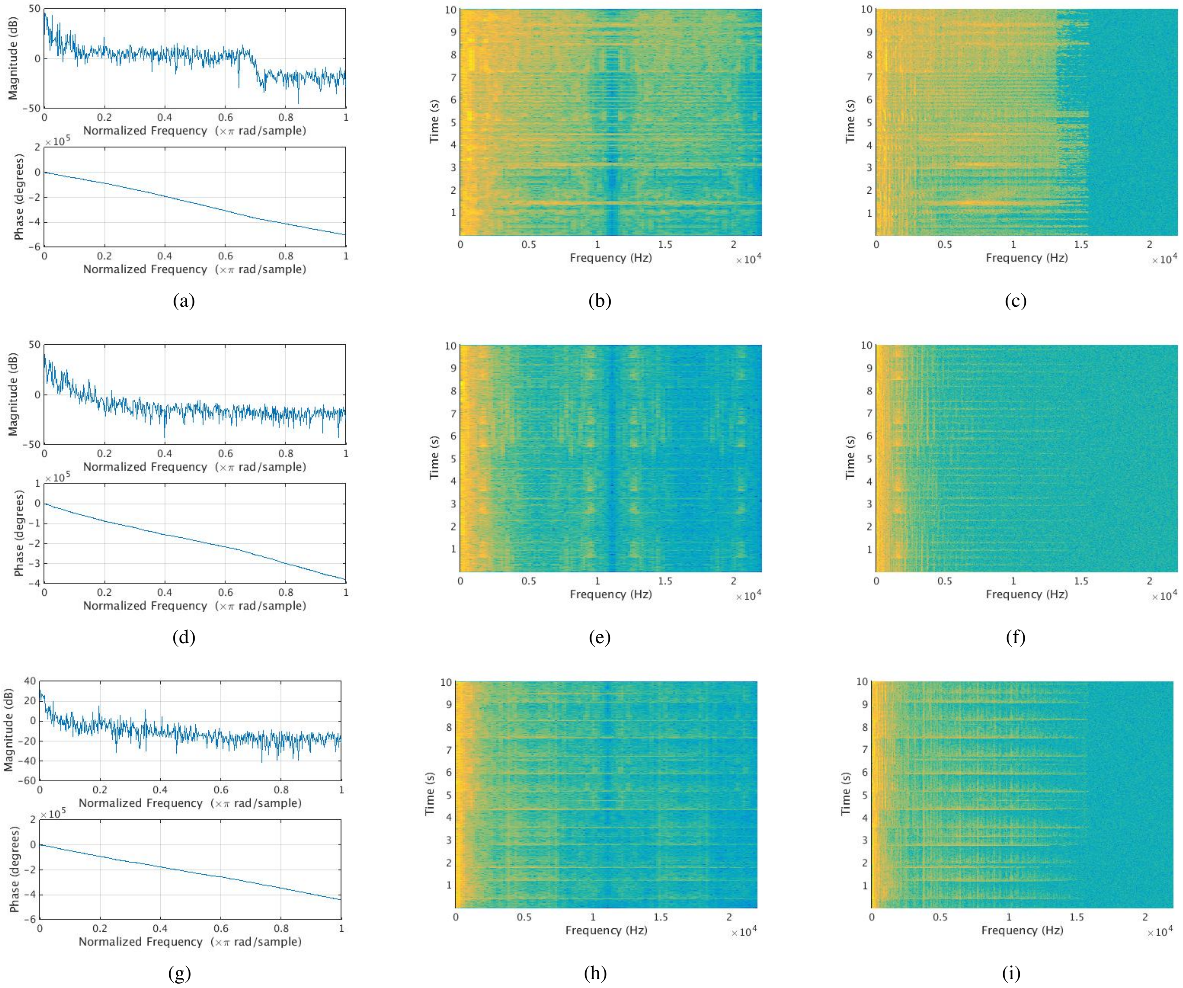
$$spectrogram(w, t) = \|STFT(w, t)\|^2$$

where t denotes the time axis of $x(t)$. The resulting spectrogram on resized audio files has been unrolled as an input matrix $M_1 \in R^{N \times L}$ where N is the number of total examples in the dataset and L is the dimensionality of unrolled power spectrogram for each signal. Section II-B discusses the dimensionality reduction on the matrix M_1 to explore the accuracy rate in different manifolds.

2) MFCC: MFCCs represent short-duration musical textures by encoding a sound's timbral information and have been shown to be effective for music genre classification [29] in the literature. In this paper, we have adopted MFCCs as the input set of hand-crafted feature vectors for SVMs and regression models to benchmark against DNNs. The process of constructing the MFCCs begins by applying a FFT to fixed

[3] Data augmentation to address data imbalances or removing small classes is part of preprocessing.

[4] FFT can be replaced with the short-time Fourier transform (STFT) in practice.



(a)

(b)

(c)

(d)

(e)

(f)

(g)

(h)

(i)

Fig. 2. A sample audio input from class I, II and, III (*alternative, pop and rock genre*) is illustrated as (a,d,g) original input signal in the frequency domain by magnitude and phase, (b,e,h) frequency spectrum and (c,f,i) power spectrogram.

size overlapping windows (Hann or Hamming windows). A series of transformations combining an auditory filter bank with a cosine transform will result in a discrete representation of each audio window in terms of MFCC descriptors [43]. Each short-time audio window is represented as an MFCC feature vector composed of 13 cepstral coefficients. Each audio sample is then represented as a $k \times 13$ feature vector where k is the number of feature vectors for each song and has been set experimentally at $k = 500$. For N total examples from the dataset, the matrix $M_2 \in R^{N \times k \times 13}$ is the input for SVMs and regression based classifiers.

B. Manifold Learning

Manifold learning is a non-linear dimensionality reduction method and is often applied in large scale datasets to avoid the curse of dimensionality [27] and reduce the time complexity of the learning models on high dimensional input. It is typically assumed that the original high-dimensional data lies on an embedded non-linear manifold within the higher-dimensional space.

Let matrix $M_1 \in R^{N \times L}$ represent the input matrix of all preprocessed examples in our dataset as described in Section II-A1, where each row represents an example unrolled preprocessed audio signal. Once again N represents the total number of signals and L is the dimensionality of the unrolled power spectrogram of input audio signals (with and without wavelet filtering for different experiments). If the attribute or representation dimensionality L is large, then the space of unique possible states is exponentially large and sampling the space becomes difficult suffering from very high time complexity (curse of dimensionality [27]). This motivates a set of dimensionality reduction and manifold learning techniques to map the original data points in a lower-dimensional space. In this paper, we experiment with principal component analysis (PCA) [24], supervised locality preserving projections (SLPP) [41] and spectral regression (SR) [10] as dimensionality reduction and manifold learning techniques. The $M_{DR} \in R^{N \times j}$ represents the reduced dataset with different models and serves as the input to DNNs, where $j \ll L$.

C. Deep Neural Networks

Our proposed DNN architecture is based on a two hidden layered feed forward neural network [20]. The initial network weights are first randomly initialized using the *Xavier initialization* [17]. In our model we select the number of neurons, $n = 400$, in each network layer with an empirical approach to boost the classification accuracy rate and minimize the final cost function of our model. The input and output relationship in the network is learned via a back-propagation where the weights W_{ij} connecting the neurons from one layer to the next, approximate a global minima. Rectified Linear Units (ReLUs) [37] have been chosen as the activation function for the DNNs and can be described as:

$$\phi(x) = max(0, x)$$

This activation function influences faster learning and is non-saturating in large scale datasets [37].

We utilize the cross entropy cost function. The cross-entropy function forces a penalty for wrong labels l_i during the learning process and avoids the local minimum convergence for the neural network. The cross entropy cost function is defined as:

$$Cost_{entropy} = -\sum_i log(\frac{e^x}{\sum_i e^x}) \times t_i$$

for input x and target output t_i.

The weight regularization of learnt weights is also an essential part of estimating the final cost function. In this paper we penalize the weights with a standard ℓ_2 weight regularization as follows:

$$Cost_{weighted} = C\lambda \sum |w|^2$$

where C is the regularization constant and λ is the weight penalty. The final cost function can then be described as:

$$Cost_{final} = -\sum_i log(\frac{e^x}{\sum_i e^x}) \times t_i + C\lambda \sum |w|^2$$

The output of the final $Cost_{final}$ is then back propagated throughout the network to adjust the weights. We use the well-known limited memory BFGS algorithm (L-BFGS) [31] for parameter estimation in the DNN. Similar to the classic BFGS, L-BFGS uses an estimation to the inverse Hessian matrix to steer its search through large-scale variable space in less time. The L-BFGS however stores only a few vectors that represent the approximation implicitly rather than the dense inverse hessian approximation of the BFGS. L-BFGS requires linear memory and is particularly well suited for optimization problems with a large number of variables [31]. The memory benefits of L-BFGS makes it a suitable candidate for parameter estimation in our model and hence is selected as the optimization algorithm.

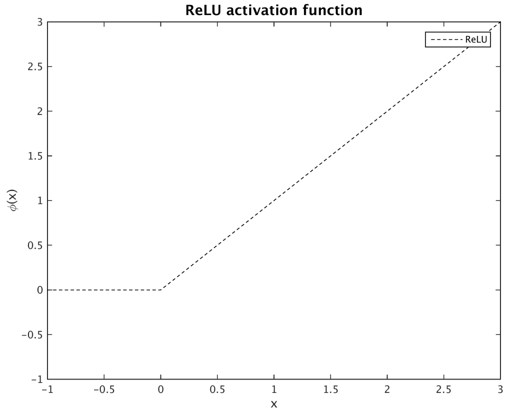


Fig. 3. The Rectified Linear units as the activation function is illustrated.

D. Baseline Classifiers

In this paper we adopt SVM, ℓ_1-SVM, logistic regression and ℓ_1-regression as the baseline classifiers for the genre classification task. The ℓ_1-SVM classifier combines the ideas of the classical SVM with the sparse approximation techniques with the goal of obtaining higher generalization accuracy on test data, increasing the robustness against overfitting to the training examples, and providing scalability in terms of the classification complexity. SVM, ℓ_1-SVM and regression-based models have been studied in the machine learning community for music genre classification [46], [2], [32] and are hence selected as baseline classifiers in this paper.

III. Experimental Results

In this section we present the results of a two hidden layered deep neural network architecture for music genre classification on a public dataset. Classification accuracy rate(%) will be used to draw a comparison between DNNs and MFCCs with baseline classifiers for music genre classification. We perform a 10-fold cross validation to evaluate the classification accuracy across all experiments.

A. Dataset

We evaluate our experimental results on a benchmark dataset for audio classification and clustering proposed by Homburg et al. [22]. The dataset contains samples of 1886 songs obtained from the Garageband site and is comprised of 9 music genres including Pop, Rock, Folk/Country, Alternative,

Genre	Number of Samples
alternative	145
blues	120
electronic	113
folkcountry	222
funksoul/rnb	47
jazz	319
pop	116
raphiphop	300
rock	504

TABLE I. Number of songs per genre.

TABLE II. CLASSIFICATION ACCURACY (%) ON SAMPLE DATASET IS COMPARED ACROSS VARIOUS METHODS.

Preprocessing	Augmentation	DR-method	Model	Average accuracy rate(%)
Raw wav	No	PCA (270 dimensions)	2-DNNs(400 neurons)	18.55%
Raw wav	No	SLPP	2-DNNs(400 neurons)	**17.03%**
Raw wav	No	SR	2-DNNs(400 neurons)	17.46%
Spectrogram	No	None	2-DNNs(400 neurons)	36.24%
Spectrogram	No	PCA (270 dimensions)	2-DNNs(400 neurons)	37.11%
Spectrogram	No	PCA (400 dimensions)	2-DNNs(400 neurons)	37.33%
Wavelet	No	PCA(400 dimensions)	2-DNNs(400 neurons)	**38.84%**
MFCC	RM-Uniform	None	ℓ_2-SVM	32.90%
MFCC	RM-Uniform	None	logistic regression	34.43%
MFCC	RM-Uniform	None	ℓ_1-SVM	37.43%
MFCC	RM-Uniform	None	ℓ_1-regression	30.45%

Jazz, Electronic, Blues, Rap/HipHop, and Funk/Soul. As illustrated in Table I, the number of samples vary by the genre. Throughout our experiments we exclude the funksoul/rnb genre due to small sample size without data augmentation and duplication. Each song is associated with a 10 seconds audio sample drawn from a random position of the corresponding song. All audio samples are encoded using mp3 format with a sampling rate of 44100Hz and a bit-rate of 128mbit/s. A total of 49 audio features including temporal features, spectral features, and some phase space features (TSPS) are also provided as part of the benchmark [36]. The MFCCs are extracted using the *Auditory Toolbox* [43] as the primary feature representations for baseline classifiers. The raw audio signals are sampled and used for deep neural networks. We report the average classification accuracy rate across all experiments.

B. Results

In this section we explain the results of our comparative experiments for music genre classification on the public dataset described in Section III-A. The first round of experiments are performed on raw audio signals using a two hidden layer neural network. All audio samples have been reduced to the have the same dimensionality as the least example size in the dataset (denoted by *RM-resize* in Table II) [5]. The top set of experiments in Table II shows the genre classification results via a two-layer deep neural network (2-DNN) with 400 neurons. The activation function for the DNNs is the ReLU function. A round of manifold learning has been applied as the dimensionality reduction step with PCA, SR and SLPP. We can observe that DNNs on the raw input audio signal perform poorly even with manifold learning. This motivates our next round of experiments with audio preprocessing.

The next round of experiments explore preprocessing of input audio signals to boost the genre classification accuracy on the dataset. The results of this set of experiments are illustrated in the middle section of Table II. The input wav files have again been resized by keeping the minimum number of columns across all examples in the dataset. The resized columns are then represented as the audio power spectrogram. The spectrogram has been derived by a 1024 DFT using a Hanning window at 44100 Hz. Rows 5, 6 and 7 show the genre classification accuracy rate on the power spectrogram using a two-layer neural network with 400 neurons. PCA with 270 and 400 principal components have been selected as the dimensionality reduction method. We can observe that the power spectrogram representation of raw input audio files can boost the classification accuracy in combination with dimensionality reduction methods. We also experiment with the power spectrogram on wavelet representation of input signals using a kaiser window as shown in row 8 of Table II. The wavelet power spectrogram representation with PCA and a 2-DNN can marginally boost the accuracy rate for genre classification on the dataset.

Finally, we compare the classification accuracy rates to hand-crafted feature extraction in combination with different classifiers. The MFCC feature vectors have been selected as the audio representation where each MFCC is a 13 dimensional feature vector. Each audio sample has been represented by 500 MFCC feature vectors selected at random. The bottom section from Table II show the results of this set of experiments on the dataset. We can observe that the preprocessing in combination with dimensionality reduction can outperform these results.

IV. CONCLUSION

Music genre classification is an interesting and hard problem with numerous applications in today's large-scale datasets and online streaming services. In this paper we explored the accuracy rate of multiple techniques for music genre classification on a public dataset. We compared the results using raw audio data and power spectrogram of input signals with a two-layer neural network. We explored different manifold learning techniques such as PCA, SLPP and SR as dimensionality reduction methods. We then compared the classification results using hand-crafted audio feature vectors such as MFCCs. Our experimental results indicate that MFCCs in combination with ℓ_1-SVM classifiers are comparable with neural networks on power spectrograms for music genre classification.

In the future, we intend to study other network architectures such as convolutional neural networks and stacked autoencoders for music classification. We will explore different signal preprocessing and representations to measure the network sensitivity for classification.

REFERENCES

[1] A. Anglade, E. Benetos, M. Mauch, and S. Dixon. Improving music genre classification using automatically induced harmony rules. *Journal of New Music Research*, 39:349–361, 2010.

[5] Zero padding can introduce additional noise and consequently decrease the accuracy of the classifiers. Hence the normalization step only resizes all samples to minimum size across all dataset points.

[2] K. Aryafar, S. Jafarpour, and A. Shokoufandeh. Automatic musical genre classification using sparsity-eager support vector machines. In *Pattern Recognition (ICPR), 2012 21st International Conference on*, pages 1526–1529. IEEE, 2012.

[3] K. Aryafar, S. Jafarpour, and A. Shokoufandeh. Music genre classification using sparsity-eager support vector machines. Technical report, Technical report, Drexel University, 2012.

[4] K. Aryafar and A. Shokoufandeh. Music genre classification using explicit semantic analysis. In *Proceedings of the 1st international ACM workshop on Music information retrieval with user-centered and multimodal strategies*, pages 33–38. ACM, 2011.

[5] K. Aryafar and A. Shokoufandeh. Fusion of text and audio semantic representations through cca. In *Multimodal Pattern Recognition of Social Signals in Human-Computer-Interaction*. Springer, 2014.

[6] K. Aryafar and A. Shokoufandeh. Multimodal music and lyrics fusion classifier for artist identification. In *Machine Learning and Applications (ICMLA), 2014 13th International Conference on*, pages 506–509. IEEE, 2014.

[7] K. Aryafar and A. Shokoufandeh. Multimodal sparsity-eager support vector machines for music classification. In *Machine Learning and Applications (ICMLA), 2014 13th International Conference on*, pages 405–408. IEEE, 2014.

[8] W. Balkema and F. van der Heijden. Music playlist generation by assimilating gmms into soms. *Pattern Recognition Letters*, 31(11):1396 – 1402, 2010.

[9] Y. Bengio. Learning deep architectures for ai. *Foundations and trends® in Machine Learning*, 2(1):1–127, 2009.

[10] D. Cai, X. He, and J. Han. Spectral regression for efficient regularized subspace learning. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8. IEEE, 2007.

[11] Z. Cataltepe, Y. Yaslan, and A. Sonmez. Music genre classification using MIDI and audio features. *EURASIP J. Appl. Signal Process.*, 2007:150–150, January 2007.

[12] L. Deng and D. Yu. Deep learning: methods and applications. *Foundations and Trends in Signal Processing*, 7(3–4):197–387, 2014.

[13] D. P. W. Ellis. Classifying music audio with timbral and chroma features. In *Proc. Int. Conf. on Music Information Retrieval ISMIR-07, Vienna, Austria*, 2007.

[14] S. Esmaili, S. Krishnan, and K. Raahemifar. Content based audio classification and retrieval using joint time-frequency analysis. In *Acoustics, Speech, and Signal Processing, 2004. Proceedings.(ICASSP'04). IEEE International Conference on*, volume 5, pages V–665. IEEE, 2004.

[15] A. Flexer, D. Schnitzer, M. Gasser, and G. Widmer. Playlist generation using start and end songs. In J. P. Bello, E. Chew, and D. Turnbull, editors, *ISMIR*, pages 173–178, 2008.

[16] Z. Fu, G. Lu, K. M. Ting, and D. Zhang. A survey of audio-based music classification and annotation. *Multimedia, IEEE Transactions on*, 13(2):303–319, 2011.

[17] X. Glorot and Y. Bengio. Understanding the difficulty of training deep feedforward neural networks. In *International conference on artificial intelligence and statistics*, pages 249–256, 2010.

[18] A. Graves, M. Liwicki, S. Fernández, R. Bertolami, H. Bunke, and J. Schmidhuber. A novel connectionist system for unconstrained handwriting recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(5):855–868, 2009.

[19] P. Hamel and D. Eck. Learning features from music audio with deep belief networks. In *ISMIR*, pages 339–344. Utrecht, The Netherlands, 2010.

[20] L. K. Hansen and P. Salamon. Neural network ensembles. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (10):993–1001, 1990.

[21] G. E. Hinton, S. Osindero, and Y.-W. Teh. A fast learning algorithm for deep belief nets. *Neural computation*, 18(7):1527–1554, 2006.

[22] H. Homburg, I. Mierswa, B. Möller, K. Morik, and M. Wurst. A benchmark dataset for audio classification and clustering. In *ISMIR*, pages 528–531, 2005.

[23] D.-N. J. Jiang, L. Lu, H.-J. Zhang, J.-H. Tao, and L.-H. Cai. Music type classification by spectral contrast feature. *Proceedings IEEE International Conference on Multimedia and Expo*, 1:113–116, 2002.

[24] M. A. Kramer. Nonlinear principal component analysis using autoassociative neural networks. *AIChE journal*, 37(2):233–243, 1991.

[25] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[26] H. Lee, P. Pham, Y. Largman, and A. Y. Ng. Unsupervised feature learning for audio classification using convolutional deep belief networks. In *Advances in neural information processing systems*, pages 1096–1104, 2009.

[27] J. A. Lee and M. Verleysen. *Nonlinear dimensionality reduction*. Springer Science & Business Media, 2007.

[28] T. Li, M. Ogihara, and Q. Li. A comparative study on content-based music genre classification. In *in Proc. SIGIR, 2003*, pages 282–289.

[29] T. L. Li and A. B. Chan. Genre classification and the invariance of mfcc features to key and tempo. In *Proceedings of the 17th international conference on Advances in multimedia modeling - Volume Part I*, MMM'11, pages 317–327, Berlin, Heidelberg, 2011. Springer-Verlag.

[30] T. Lidy and A. Rauber. Improving genre classification by combination of audio and symbolic descriptors using a transcription system. In *Proc. ISMIR*, 2007.

[31] R. Malouf. A comparison of algorithms for maximum entropy parameter estimation. In *proceedings of the 6th conference on Natural language learning-Volume 20*, pages 1–7. Association for Computational Linguistics, 2002.

[32] M. I. Mandel and D. P. Ellis. Song-level features and support vector machines for music classification. In *ISMIR 2005: 6th International Conference on Music Information Retrieval: Proceedings: Variation 2: Queen Mary, University of London & Goldsmiths College, University of London, 11-15 September, 2005*, pages 594–599. Queen Mary, University of London, 2005.

[33] C. McKay and I. Fujinaga. Musical genre classification: Is it worth pursuing and how can it be improved? In *ISMIR*, pages 101–106, 2006.

[34] C. McKay and I. Fujinaga. Combining features extracted from audio, symbolic and cultural sources. In *ISMIR*, pages 597–602, 2008.

[35] A. Meng, P. Ahrendt, and J. Larsen. Improving music genre classification by short-time feature integration. In *IEEE ICASSP*, pages 497–500, 2005.

[36] I. Mierswa and K. Morik. Automatic feature extraction for classifying audio data. *Machine Learning Journal*, 58:127–149, 2005.

[37] V. Nair and G. E. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 807–814, 2010.

[38] R. Neumayer and A. Rauber. Integration of text and audio features for genre classification in music information retrieval. In *Proceedings of the 29th European Conference on Information Retrieval*, pages 724–727, 2007.

[39] A. Y. Ng, M. I. Jordan, Y. Weiss, et al. On spectral clustering: Analysis and an algorithm. *Advances in neural information processing systems*, 2:849–856, 2002.

[40] N. Scaringella, G. Zoia, and D. Mlynek. Automatic genre classification of music content: a survey. *Signal Processing Magazine, IEEE*, 23(2):133–141, 2006.

[41] C. Shan, S. Gong, and P. W. McOwan. Appearance manifold of facial expression. In *Computer Vision in Human-Computer Interaction*, pages 221–230. Springer, 2005.

[42] S. Sigtia and S. Dixon. Improved music feature learning with deep neural networks. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pages 6959–6963. IEEE, 2014.

[43] M. Slaney. Auditory toolbox, version 2. Technical Report 1998-10, Interval Research Corporation, Palo Alto, California, USA, 1998.

[44] G. Tzanetakis and P. R. Cook. Musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing*, 10(5):293–302, 2002.

[45] B. Whitman. Combining musical and cultural features for intelligent style detection. In *Proc. Int. Conf. Music Information Retrieval (ISMIR)*, pages 47–52, 2002.

[46] Y.-H. Yang, Y.-C. Lin, Y.-F. Su, and H. H. Chen. Music emotion classification: A regression approach. In *Multimedia and Expo, 2007 IEEE International Conference on*, pages 208–211. IEEE, 2007.